

IN THE SPECIFICATION

Amendments to the Abstract:

Please amend the Abstract as follows:

~~This invention is.~~ The embodiments herein are concerned with building an information system for enterprises. It has a unique versatile executor engine, which can interpret and execute transaction structures and information views. The information system ~~thus built~~ can grow along with the needs of the enterprise. ~~The novelty lies in the new methodology that has been evolved to build information systems with the executor engine.~~ An information system built in this methodology will allow online and on site creation of new options and changing the existing options without any down time. ~~Profit SRM incorporates.~~ The embodiments herein incorporate a unique technology, which enables an information system to be built without writing software programs. This makes information system building simple and quick. The data is analyzed and split into transaction structures and information views. These are then defined using the ~~PROFIT SRM~~ builder, which stores these definitions as data. This is intercepted and executed by the unique executor engine to provide the necessary information. ~~The architecture of the system is described in FIG. 1.~~

Amendments to the Specification:

Please amend the following at paragraphs [0041-0046] of the specification as follows:

FIG. 1 of the drawing shows the open-ended architecture of a tool for building an information system. ~~PROFIT SRM~~

FIG. 2 of the drawing shows the development cycle according to prior art.

FIG. 3 of the drawings shows the development cycle according to one embodiment. ~~invention.~~

FIG. 4 of the drawings shows the time frame for building an information system according to one embodiment. ~~in a conventional method.~~

FIG. 5 of the drawings shows the time frame for building an information system with products developed in the prior art.

FIG. 6 of the drawings shows the time frame for building an information system with products developed in the prior art, using PROFIT 5 RM.

Please amend the following at paragraphs [0048-0055] of the specification as follows:

Refer to FIG. 1 for the open ended architecture diagram of ~~PROFIT 5RM~~ a tool for building an information system 100.

A domain expert 110 interacts with the end user 120 and understands the information system required by the organization.

The information system is analyzed and made into simple structures called Transaction Structures and Information Views. These are defined using the ~~PROFIT 5RM~~ Builder 130.

The Builder 130 stores the definitions of Transaction Structures and Information View as data into the database 150.

The Graphical User Interface Layer (GUIL) 160 presents a user interface for the end user 120 based on the Transaction Structure.

The GUIL 160 interacts with the end user and ~~submits-submit~~ various requests for transaction processing or information request to the Process Request Server [PRS] 170.

Process Request Server (PRS) 170 provides services to process transactions and information requests. The Transaction Structures govern the transaction processing done by the PRS 170. The Information Views govern the processing of information requests.

In this approach, software programs are not written to make an information system. Instead, an information system is made into Transaction Structures and Information View and stored in the database 150.

Please amend the following at paragraph [0064] of the specification as follows:

The executor engine 140 interprets a Transaction Structure and a transaction form is presented to the end user 120. The end user 120 can input data into the information system through these forms.

Please amend the following at paragraphs [0103-0107] of the specification as follows:

FIG. 1 of the drawing shows the open-ended architecture of ~~PROFIT-SRRM~~ a tool for building an information system 100. FIG. 2 of the drawings shows the development cycle 200 as per the prior art, which includes scoping 210, requirement study 220, system design, database design 230, program design 240, coding 250, testing 260, training 270 and implementation 280.

The software executor engine invented by us will cut short the software development cycle to a great extent. Programming efforts will be removed completely from the development cycle. The time saving could be described as follows: What can be done by an experienced programmer in a month, can be done by a non-programmer in just a few hours. The time saving ratio is 1:50. FIG. 3 of the drawings shows the development cycle 300 with ~~PROFIT-SRM~~ the tool 100 which includes scoping 310, requirement study 320, solution definition document preparation 330, solution definition 340, testing 350, training 360 and implementation 370.

FIG. 4 shows the time frame 400 for building an information system in conventional method. The methodology includes development of the modules for a small manufacturing enterprise right from system study, design & complete coding. The basic modules ~~covered~~ covered are financial accounting, inventory, production planning and control, order processing and payroll.

FIG. 5 shows in detail the time frame 500 for building an information system with products developed in the prior art. This methodology includes customization of basic existing modules for a small manufacturing enterprise. The enhancements could be achieved only by modifying the program. The basic modules covered are financial accounting, inventory, production planning and control, ~~control~~, order processing and payroll.

FIG. 6 shows the time frame 600 for building an information system with products developed in the prior art, using PROFIT-SRM. This methodology includes definition of the complete requirements for a small manufacturing enterprise without programming. The enhancements could be achieved by merely modifying the definitions but not coding. The basic modules covered are financial accounting, inventory, production planning and control, order processing and payroll.